

In the claims

1. (Currently Amended) A client-server computer system comprising:
 - at least one client application server having an application, including configuration variables, that configures the application for operation;
 - an application properties server accessible by at least one of said ~~at least one~~ client application servers, said application properties server coupled to ~~said~~ at least one said client application server for providing configuration variable data in response to a request from ~~said~~ at least one said client application server, wherein the request is provided from the at least one client application server to the application properties server via a command line interface that includes one or more switches to control how the requested data is returned, the one or more switches including at least a hashkey switch specifying that the configuration variables are to be returned in a hashtable;
 - an administration system coupled to said application properties server;
 - and
 - a storage medium coupled to said application properties server and said administration system for centrally storing said configuration variable data.
2. (Currently Amended) A client-server computer system comprising:
 - a plurality of client application servers operating multiple computer network protocols, each client application server having an application including configuration variables that configure the application for operation;
 - an application properties server network accessible by said plurality of client application servers via at least one application software protocol, wherein said application server network provides configuration variable information in response to at least one configuration request from at least one client application server; and,
 - a storage medium coupled to said application server network, wherein said storage medium for storing stores updated system current-configuration variable information, wherein said storage medium and is administered by a server coupled to said client-server system.

3. (Original) A client-server computer system according to claim 2, wherein said storage medium comprises a database.
4. (Original) A client-server computer system according to claim 2, wherein said configuration information is represented by a storage schema in the form of Lightweight Directory Access Protocol.
5. (Original) A client-server computer system according to claim 3, wherein said database contains a table-based system of configuration information, wherein said tables are searchable by said application server network in response to a request from at least one client.
6. (Original) A client-server computer system according to claim 4, wherein the storage schema represented by Lightweight Directory Access Protocol represents a table-based system configuration information.
7. (Original) A client-server computer system according to claim 2, wherein said database stores configuration information that is dynamically updateable by an external administrator.
8. (Original) A client-server computer system according to claim 2, wherein said storage schema is in the form of Lightweight Directory Access Protocol and represents configuration information that is dynamically updateable by an external administrator.
9. (Original) A client-server computer system according to claim 7, wherein at least one client is coupled to said application server network via an RMI interface.
10. (Original) A client-server computer system according to claim 8, wherein at least one client is coupled to said application server network via an RMI interface.

11. (Original) A client-server computer system according to claim 3, wherein said configuration information is stored and retrieved from said storage medium via Key Value Pairs.

12. (Original) A client-server computer system according to claim 4, wherein said configuration information is stored and retrieved from said storage medium via Hashtable Hierarchy.

13. (Original) A client-server computer system according to claim 7, wherein said configuration information is stored and retrieved from said storage medium via Key Value Pairs.

14. (Original) A client-server computer system according to claim 8, wherein said configuration information is stored and retrieved from said storage medium via Hashtable Hierarchy.

15. (Currently Amended) An application properties server network comprising:
a plurality of client application servers operating applications using a plurality of computer protocols and requiring configuration variable data to configure the applications for operation;

means for performing configuration services in response to configuration requests from said plurality of client application servers, said means for performing configuration services being coupled to said plurality of client application servers;

means for storing and maintaining a system of configuration variable information data coupled to said means for performing configuration services; and

means for interfacing said plurality of client application servers to said means for performing configuration services.

16. (Original) An application server according to claim 15, wherein said means for interfacing said plurality of client application servers to said means for performing configuration services includes a CORBA server application.

17. (Original) An application server according to claim 16, wherein said means for interfacing said plurality client application servers to said means for performing configuration services includes a properties server application for handling RMI requests for configuration services.
18. (Original) An application server according to claim 17, wherein said means for interfacing said plurality of client application servers to said means for performing configuration services includes a common database access library.
19. (Original) An application server according to claim 18, wherein said means for interfacing said plurality of client application servers to said means for performing configuration services includes a database server coupled to said properties server application for handling RMI requests and said CORBA server application for interacting with said means for storing and maintaining configuration information.
20. (Original) An application server according to claim 15, further comprising a Java RMI Application Programming Interface.
21. (Original) An application server according to claim 20, further comprising a CORBA gateway.
22. (Original) An application server according to claim 21, wherein said means for performing configuration *service* is implemented by a base Java RMI service in a service broker framework.
23. (Original) An application server according to claim 22, wherein said service broker framework is implemented using at least one XML service broker configuration file.
24. (Original) An application server according to claim 23, wherein said configuration information is stored and retrieved from said means for storing via Key Value Pairs.

25. (Original) An application server according to claim 23, wherein said configuration information is stored and retrieved from said means for storing via Hashtable Hierarchy.

26. (Currently Amended) A system for providing an application configuration service, the system comprising:

an application properties server;

at least one ~~application running a~~ Java application program, including application variables that configure the application for operation, and networked with said application properties server;

at least one ~~application running a~~ CORBA application program, including application variables that configure the application for operation, and networked with said application properties server;

at least one ~~application running an~~ Internet application program, including application variables that configure the application for operation, and networked with said application properties server;

one or more application programming interfaces, ~~the one or more application programming interfaces~~ coupled to the application properties server for receiving data configuration service requests via a plurality of computer network protocols;

~~and~~ at least one dynamically-maintainable configuration variable data schema coupled to said application properties server.

27. (Original) A client-server computer system according to claim 26, wherein said data schema comprises configuration information and is at least partially in the form of a database.

28. (Original) A client-server computer system according to claim 26, wherein said data schema comprises configuration information in the form of Lightweight Directory Access Protocol.

29. (Original) A client-server computer system according to claim 27, wherein said application server and said data schema are remotely located to said plurality of client

application servers and said configuration information is maintainable by a remote administrator.

30. (Original) A client-server computer system according to claim 28, wherein said application server and said data schema are remotely located to said plurality of client application servers and said configuration information is maintainable by a remote administrator.

31. (Original) The client-server computer system of claim 29, wherein data passes between said application servers and said application server in the form of a string.

32. (Original) The client-server computer system of claim 30, wherein data passes between said application servers and said application server in the form of a string.

33. (Original) The client-server computer system of claim 29, wherein data passes between said application servers and said application server in the form of a hashtable.

34. (Original) The client-server computer system of claim 30, wherein data passes between said application servers and said application server in the form of a hashtable.

35. (Currently Amended) A server system for providing ~~data~~ configuration services in response to requests from applications coupled to the server, the system comprising:

a CORBA application server running application program including application variables that configure the application for operation;

an RMI application server running application program including application variables that configure the application for operation;

an internet application server running application program including application variables that configure the application for operation,

wherein ~~said~~ the application servers are all in communication with a centralized application properties server, the application servers providing configuration requests to the application properties server via a network;

one or more application programming interfaces, ~~the one or more application programming interfaces~~ capable of handling a plurality of software protocols in communication with the application properties server and ~~said the~~ application servers; and

a configuration variable data schema in communication with said network, for storing configuration variable[[s]] data and accessible by said properties server.

36. (Currently Amended) A server system according to claim 35, wherein said data schema comprises an ~~ORACLE~~ relational database.

37. (Original) A server system according to claim 35, wherein said configuration variables comply with Lightweight Directory Access Protocol.

38. (Original) A server system according to claim 35, wherein said one or more interfaces includes a CORBA server application.

39. (Original) A server system according to claim 38, wherein said one or more interfaces includes a server application for handling RMI requests.

40. (Original) A server system according to claim 39, wherein said one or more interfaces includes a common database access library.

41. (Original) A server system according to claim 40, wherein said one or more interfaces includes a database server coupled to said properties server.

42. (Original) A server system according to claim 35, further comprising a Java RMI Application Programming Interface.

43. (Original) A server system according to claim 42, further comprising a CORBA gateway.

44. (Original) An server system according to claim 43, wherein said properties server is implemented by a base Java RMI service in a service broker framework.

45. (Original) A server system according to claim 44, wherein said service broker framework is implemented using at least one XML service broker configuration file.

46. (Currently Amended) A system for providing an application configuration service, the system comprising:

means for receiving a configuration service request from a ~~customer~~ client running an application requiring configuration variable data to configure the application for operation, the client ~~customer~~ requesting configuration variable information from said system;

means for sending a configuration service request instruction to an application server corresponding to the configuration variable information;

means for sending a configuration service request from the application server to a relational database, the configuration service response being based at least in part on the configuration service request;

means for performing configuration services in response to ~~said the~~ configuration service request;

means for updating ~~said the~~ relational database based on current configuration data requirements of ~~said the~~ system;

means for sending a configuration result from the application server to ~~said the customer~~ client based at least in part on the configuration service request; and

means for providing a response to ~~said the~~ system from ~~said the client customer to said system in response~~ acknowledging to said the configuration result.

47. (Currently Amended) A computer-readable medium storing a plurality of instructions adapted to be executed by a processor for providing an application configuration service, the plurality of instructions comprising instructions to:

receive a configuration service request from a ~~customer data~~ client device running an application requiring configuration variable data to configure the application for

~~operation, the customer data device including an instruction concerning configuration files;~~

~~generate a service session instruction, the service session instruction based at least in part on the service request;~~

~~send the service session instruction to one or more open application programming interfaces, the service session instruction corresponding to one or more ~~data~~-configuration variable data requests from ~~said the client customer data~~ device;~~

~~perform one or more configuration functions based on stored ~~variables~~ configuration variable data in a relational database; and~~

~~send a configuration service response to the client ~~customer data~~ device, the ~~configuration service response~~ based on the configuration service request.~~

48. (Currently Amended) A medium according to claim 47, wherein said relational database comprises ~~an Oracle~~ a relational database and further comprises an instruction to load at least a portion of said database into a memory upon startup of said application service.

49. (Original) A medium according to claim 47, wherein said variables are stored in the format of Lightweight Directory Access Protocol and further comprise an instruction to load variables into a memory upon startup of said application service.

50. (Currently Amended) A method of providing ~~[[a]]~~ configuration services ~~with to~~ a client-server computer system comprising the steps of:

~~coupling a configuration service request between a client application server and an application properties server, wherein the configuration service request contains a plurality of configuration service request instructions to update the configuration variable data required to configure the application for operation on the client application server;~~

~~providing delivering ~~[[a]]~~ the configuration service request instructions to a data schema in response to said configuration service request coupled between said client application server and said application server;~~

retrieving updated configuration data files from a storage mass coupled to said application properties server; and
coupling a response to said client application server.

51. (Currently Amended) A method for providing an application service, ~~the method~~ comprising:

a step for sending a ~~data~~ configuration service request from a user wherein the configuration service request contains a plurality of configuration service request instructions to update configuration variable data required to configure the application for operation on a client application server;

a step for generating a configuration service instruction, ~~the service instruction~~ based at least in part on a configuration service request from the user;

a step for sending ~~said the~~ configuration service instruction to one or more configuration variable data schemas via one or more application programming interfaces; ~~the service instruction~~ corresponding to one or more configuration variable data requests from the user;

a step for dynamically updating a table of configuration variable data files stored in ~~said the one or more configuration variable~~ data schemas based on changes to the application service;

a step for calling up at least one table of configuration variable data files from ~~the said one or more configuration variable~~ data schemas;

a step for providing configuration variable data ~~information~~ to said user in accordance with updated configuration variable data files stored in ~~the said one or more configuration variable~~ data schemas; and

a step for sending a service response to the user ~~requesting~~ initiating the configuration service request.

52. (Original) A method according to claim 51, further comprising the step of initializing at least one property server object.

53. (Original) A method according to claim 52, further comprising the step of creating a connection between said at least one property server object and at least one of said data schemas.

54. (Original) A method according to claim 53, further comprising the step of pooling at least one server object with at least one client application server.

55. (Original) A method according to claim 51, further comprising the steps of establishing a service broker framework by initializing a plurality of configurable property server objects and said objects to a plurality of client application servers.

56. (Original) A method according to claim 55, further comprising the steps of reading an XML configuration file and establishing a service for at least one service tag in said XML configuration file.

57. (Original) A method according to claim 51, further comprising the step of handling configuration files using Key Value Pairs.

58. (Original) A method according to claim 51, further comprising the step of handling configuration files using Hashtable Hierarchy.

59. (Original) A method according to claim 51, further comprising retrieving configuration files through Java Database Connectivity API.

60. (Currently Amended) A method for providing an application configuration service, ~~the method~~ comprising:
a step for sending at least one ~~data~~-configuration service request from a user ;
a step for generating at least one configuration service instruction, ~~the service~~
~~instruction~~-based at least in part on at least one configuration service request from ~~said a~~ a
user wherein the configuration service request instructions update the configuration

variable data required to configure the application for operation on a client application server;;

a step for sending at least one ~~the~~ service instruction to ~~one or more~~ at least one configuration variable data storage ~~schemas~~ schemas via ~~one or more~~ at least one application programming interfaces, ~~the service instruction corresponding to one or more configuration data requests from the user;~~

a step for dynamically updating at least one table of configuration variable data files stored in ~~said one or more~~ at least one data schemas based on changes to the application configuration service;

a step for calling up at least one table of configuration data files from ~~said one or more~~ at least one data storage schemas;

a step for providing configuration variable data information to ~~said the~~ user ~~in accordance with updated configuration data files stored in said the storage schemas;~~ and

a step for sending service response to the user initiating the application configuration service.

61. (Original) A method according to claim 60, further comprising the steps of:

establishing a service broker framework;

initializing a plurality of configurable property server objects; and

pooling said plurality of configurable property server objects to at least one client application server.

62. (Original) A method according to claim 61, further comprising the steps of:

reading an XML configuration file; and

establishing a service for at least one service tag in said XML configuration file.